KernelCare

CHEF

KernelCare: Live Kernel Patching for Linux

And how to deploy it automatically

The kernel is the most important part of any Linux system. It provides vital low-level functions to the entire system. Any security issues detected within it jeopardize the whole server, which in turn puts your customers and your revenue stream at risk.

Summary

What is KernelCare?

KernelCare is patch management software that automatically keeps your Linux kernel up to date with the latest security patches.
No server rebooting or system downtime is necessary. It is fast, simple and easy to deploy, and can deliver complex patch configurations or customized kernels without affecting performance. It is available for all major Linux distributions.

Why is it needed?

Linux has a long history of solid dependability, but like most modern operating systems, it is a large body of complex software that needs frequent updates. These updates often target perceived security weaknesses, which, if not resolved, can be exploited to compromise or debilitate your servers and data.

For example, there were over 170 Linux kernel vulnerabilities detected last year[1], some of which are fixed by individual patches. It is not uncommon for a Linux system to need monthly updates and reboots.

There is a time lag between the detection of a vulnerability and its resolution by a patch update.

This offers an unavoidable window of opportunity for malicious threat agents within which to target systems and exploit vulnerabilities.

However, once a patch is released, its effectiveness in preventing attack is severely curtailed if the patch is not immediately applied. This entirely avoidable situation is where KernelCare comes in. It virtually eliminates the gap between patch issue and patch application, by installing patches automatically and without disruption to your core services.

About KernelCare

Our team consists of expert kernel developers whose primary role is to watch for kernel vulnerabilities and prepare patches for them. These are released as soon as possible, often much sooner than most Enterprise Linux vendor releases. We can do this, quickly, because our sole focus is on kernel security, and none of its other functionalities—we do not touch any kernel ABIs (Application Binary Interfaces).

The traditional way of patching kernels can cause unwanted or undetected functional changes to your kernel. It may even introduce new or unknown security vulnerabilities. It can also change your kernel version, triggering security alerts or necessitating full regression testing of hosted applications.

All patch updates are fully auditable - all can be selectively pre-tested and approved for distribution and installation or abandoned and rolled back. This can be done at any time with zero impact.

[1] https://www.cvedetails.com/product/47/Linux-Linux-Kernel.html?vendor_id=33

How It Works

KernelCare runs as a service that applies security updates to a running Linux kernel. A small agent installed on a server or device applies binary kernel patches. These are downloaded directly from our repository, the main KernelCare Patch Server at http://patches.kernelcare.com. This server can be accessed directly or through a firewall (via a proxy server), or a local patch update server can be self-hosted to deliver patches.

Patches are distributed as cumulative binary packages, custom-built for each supported kernel version, and each is GPG-key signed for security.

When a patch is applied with KernelCare, a reboot of the system is not required. This is not the case when using traditional update tools (e.g. yum, apt-get). Instead, the Linux kernel is binary patched, in memory. Nothing else is touched, so there is no need to update system libraries or packages to keep in step with kernel changes. In fact, the official patch level does not change (see Security Compliance).

Deployment

Automated Deployment and Remediation Management

Chef Infra can be used to automate the deployment of KernelCare to achieve the following:

- Distribute the KernelCare agent package (only necessary for servers with no internet access).
- Distribute the KernelCare agent configuration file /etc/sysconfig/kcare/kcare.conf).
- Set environment variables.
- Install the KernelCare agent (from either local or remote download servers).
- Register KernelCare.

Chef InSpec can identify vulnerabilities requiring attention, Chef Infra can then prioritize, schedule, and manage the remediation, and KernelCare under the management of Chef Infra can execute the remediation by applying security patches while your systems are operational. Full automation for a more secure environment, with no downtime.

For more details on automating KernelCare deployment with Chef, see https://docs.kernelcare.com/kernelcare_enterprise/#deployment-automation

Security Compliance

Because KernelCare patches the kernel directly in memory, the official patch identification does not change. In other words, neither the output of uname -r nor the contents of the file /proc/version change when patched.

We do this because glibc and other libraries relying on the kernel ABI (Application Binary Interface) must know the exact version of the kernel.

Although this approach provides the highest levels of stability and compatibility for servers, it can cause some security scanners to report the active kernel as 'out of date'.

To prevent such reports, KernelCare has a command that returns the effective version of the kernel.

```
kcare-uname -r
```

Security Scanner Interface

Commonly used security scanners can obtain the list of CVEs patched by KernelCare even though the output of **uname -r** stays unchanged. KernelCare agent can manipulate the kernel version as reported by DEB- and RPM-based distributions. The kernel package version output can be overridden by setting **LD_PRELOAD**. It changes the information shown by the package manager similar to:

```
[centos@host ~]$ rpm -q kernel-headers
kernel-headers-3.10.0-693.17.1.el7.x86_64
[centos@host ~]$ LD_PRELOAD=/usr/libexec/kcare/kpatch_package.so rpm -q
kernel-headers-3.10.0-957.21.3.el7.x86_64
```

Scanner interface changes system functions to rely on **kcarectl --uname** output thus making KernelCare “effective version” come into play. This behavior applies only to a single system user that should be used to run a security scan over SSH.

Enabling KernelCare Scanner Interface

The installation command looks like:

```
curl -s -L https://kernelcare.com/installer | KCARE_SCANNER_USER=username bash
```

To update an existing package, run (for RPM-based systems):

```
KCARE_SCANNER_USER=username yum update kernelcare
```

or (for DEB-based):

```
KCARE_SCANNER_USER=username apt-get update kernelcare
```

Where username is the user which will be used to run scanners on the server. Start a new SSH session for KCARE_SCANNER_USER and apply KernelCare patches (kcarectl --update). The output of installed kernel version as seen by the system package manager (RPM/DPKG) will change to the "effective version" provided by KernelCare.

New security scan results should not display any kernel-related CVEs that are covered by KernelCare binary patches.

Automated Linux kernel security updates without reboots

TRY FOR FREE

KernelCare website: https://www.kernelcare.com

KernelCare Blog: https://www.blog.kernelcare.com

KernelCare Patch Server: http://patches.kernelcare.com

KernelCare documentation: http://docs.kernelcare.com

CloudLinux Network - CLN (Billing Portal): https://cln.cloudlinux.com

CloudLinux 24/7 online support system: https://cloudlinux.zendesk.com